

# **CLAIMS**

What is claimed is:

- 1 1. A computer-implemented method for analysis of executable program code, the  
2 executable program including segments of code that correspond to callable functions in  
3 source code from which the executable code was generated, comprising:  
4 reading from the executable program code pairs of entry points and endpoints,  
5 each pair including an entry point and an endpoint that are associated with a callable  
6 function in the source code and corresponding to a segment of the executable program  
7 code; and  
8 generating analysis data for the functions identified by the pairs of entry points  
9 and end points.  
  
1 2. The method of claim 1, further comprising scanning the executable program  
2 code for selected characteristics using the pairs of entry points and endpoints.  
  
1 3. The method of claim 1, further comprising:  
2 executing the program code;  
3 detecting execution of the functions using the pairs of entry points and  
4 endpoints; and  
5 recording selected execution characteristics of each executed function.  
  
1 4. The method of claim 1, wherein the executable program code includes one or  
2 more dynamic load modules, the method further comprising:  
3 reading entry points of initializer and deinitializer functions from dynamic load  
4 modules;  
5 pairing the entry points of the initializer and deinitializer functions with  
6 endpoints of the initializer and deinitializer functions; and  
7 generating analysis data for the initializer and de-initializer functions identified  
8 by the pairs of entry points and end points of the initializer and deinitializer functions.

10005460-1

- 1 5. The method of claim 4, wherein the executable program code includes a  
2 procedure lookup table (PLT) table associated with the one or more dynamic load  
3 modules, the method further comprising:  
4 reading function entry points from the PLT;  
5 pairing the entry points from the PLT with endpoints; and  
6 generating analysis data for the PLT functions identified by the pairs of entry  
7 points and end points of the PLT functions.
- 1 6. The method of claim 4, further comprising scanning the executable program  
2 code for selected characteristics using the pairs of entry points and endpoints.
- 1 7. The method of claim 4, further comprising:  
2 executing the program code;  
3 detecting execution of the functions using the pairs of entry points and  
4 endpoints; and  
5 recording selected execution characteristics of each executed function.
- 1 8. The method of claim 4, wherein the program code includes a symbol table  
2 identifying one or more function entry points, the method further comprising:  
3 reading entry points of functions from the symbol table;  
4 pairing the entry points from the symbol table with endpoints; and  
5 generating analysis data for the symbol table functions identified by the pairs of  
6 entry points and end points of the symbol table functions.
- 1 9. The method of claim 1, wherein the program code includes a symbol table  
2 identifying one or more function entry points, the method further comprising:  
3 reading entry points of functions from the symbol table;  
4 pairing the entry points from the symbol table with endpoints; and  
5 generating analysis data for the symbol table functions identified by the pairs of  
6 entry points and end points of the symbol table functions.

10005460-1

1 10. The method of claim 1, further comprising:  
2 detecting function calls at runtime;  
3 finding the entry point of a runtime-detected function call;  
4 pairing an endpoint with the entry point of a runtime-detected function call; and  
5 generating analysis data for functions identified by pairs of entry points and end  
6 points of the runtime-detected function calls.

1 11. The method of claim 10, further comprising:  
2 detecting execution of stub functions at runtime; and  
3 bypassing analysis of stub functions.

1 12. The method of claim 1, further comprising:  
2 detecting execution of stub functions at runtime; and  
3 bypassing analysis of stub functions.

1 13. The method of claim 10, wherein the executable program code includes one or  
2 more dynamic load modules, the method further comprising:  
3 reading entry points of initializer and deinitializer functions from dynamic load  
4 modules;  
5 pairing the entry points of the initializer and deinitializer functions with  
6 endpoints of the initializer and deinitializer functions; and  
7 generating analysis data for the initializer and de-initializer functions identified  
8 by the pairs of entry points and end points of the initializer and deinitializer functions.

1 14. The method of claim 13, wherein the executable program code includes a  
2 procedure lookup table (PLT) table associated with the one or more dynamic load  
3 modules, the method further comprising:  
4 reading function entry points from the PLT;  
5 pairing the entry points from the PLT with endpoints; and  
6 generating analysis data for the PLT functions identified by the pairs of entry  
7 points and end points of the PLT functions.

10005460-1

- 1 15. An apparatus for analysis of executable program code, the executable program
- 2 including segments of code that correspond to callable functions in source code from
- 3 which the executable code was generated, comprising:
- 4 means for reading from the executable program code pairs of entry points and
- 5 endpoints, each pair including an entry point and an endpoint that are associated with a
- 6 callable function in the source code and corresponding to a segment of the executable
- 7 program code; and
- 8 means for generating analysis data for the functions identified by the pairs of
- 9 entry points and end points.